

ESTRUCTURAS DE DATOS

Curso 2018/19

PRÁCTICA 4

Cadenas de nodos enlazados

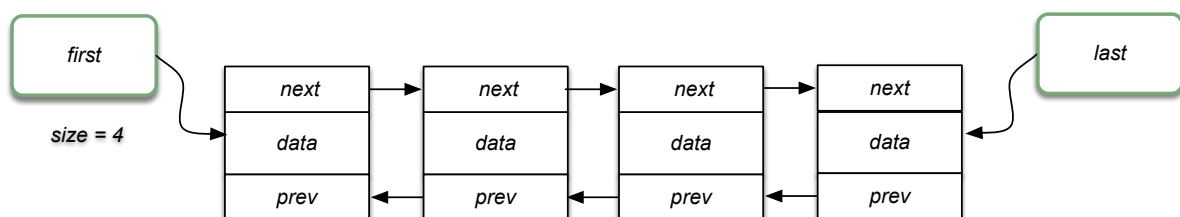
Instrucciones

- Se debe completar en una sesión.
- Práctica individual.
- Lee el enunciado completo antes de comenzar. Los comentarios incluidos en el código también pueden proporcionar información útil.
- Se habilitará una tarea para que entregues el código desarrollado.
- La práctica será APTA si se superan todos los test de validación proporcionados.

Tenemos una clase **EDDDoubleLinkedList<T>** que implemente la interface **List<T>** de la *Java Collection Framework*. En ella, decidimos almacenar los datos de la lista usando una cadena no circular de nodos doblemente enlazados. La definición de su parte privada es:

```
public class EDDoubleLinkedList<T> implements List<T> {  
    private class Node {  
        private T data;  
        private Node next;  
        private Node prev;  
  
        public Node(T data) { this.data = data; }  
    }  
  
    private Node first = null;  
    private Node last = null;  
    private int size = 0;  
}
```

- **Node**: clase privada que representa un nodo con doble enlace
- **first**: referencia al primer nodo de la cadena
- **last**: referencia al último nodo de la cadena.
- **size**: número de nodos en la cadena.



Ejercicio 1

Añade un método **reverse** a la clase que invierta el orden de los elementos de la lista. Es decir, que cambie el orden de todos sus elementos de forma que los que estén al principio acaben al final y viceversa.

```
public void reverse()
```

Ejemplo:

- [5, 6, 3, 5, 3, 9, 1, 0] \Rightarrow [0, 1, 9, 3, 5, 3, 6, 5]

Ejercicio 2

Añade un método **shuffle** a la clase. El método tomará otra lista como parámetro y la mezclará con lista local **this**, modificándola. De tal forma que el primer elemento de lista se insertará entre el primero y el segundo de la lista local, el segundo elemento de lista entre el segundo y el tercero de la lista local, y así sucesivamente. Si la lista local fuese demasiado corta se insertarían los elementos sobrantes de lista al final de la lista local. En el caso de que lista sea **null** no se hará nada

```
public void shuffle(List<T> lista)
```

Ejemplos:

- this: [9, 8, 7, 6]; lista: [2, 4, 5]; \Rightarrow this: [9, 2, 8, 4, 7, 5, 6]
- this: [99, 100]; lista: [2, 4, 5]; \Rightarrow this: [99, 2, 100, 4, 5]

NOTA: La clase **EDDLinkedList** que se te proporciona tan sólo implementa **los** métodos **size**, **isEmpty**, **clear**, **toArray**, **toString** y un constructor de copia. Si necesitas alguno de los otros métodos, deberás implementarlos.